

Data Distribution & Processing CSCI
Data Health CSC
Requirements Design Panel 3

July 1, 1997
Version 1.0

1. Data Distribution & Processing CSCI

The Data Distribution & Processing CSCI is composed of the following CSCs:

Data Distribution CSC, Data Fusion CSC, and Data Health CSC.

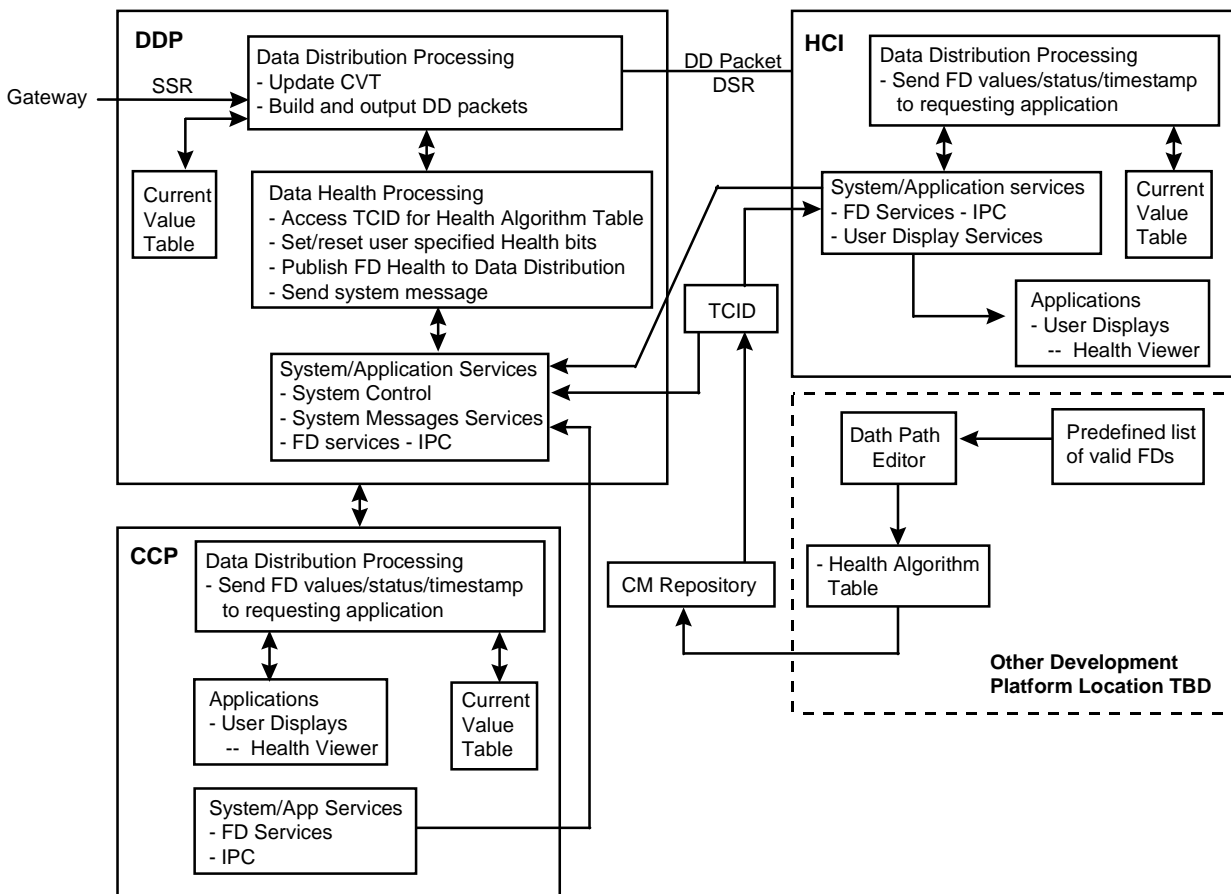
1.1 Data Health CSC Introduction

1.1.1 Data Health CSC Overview

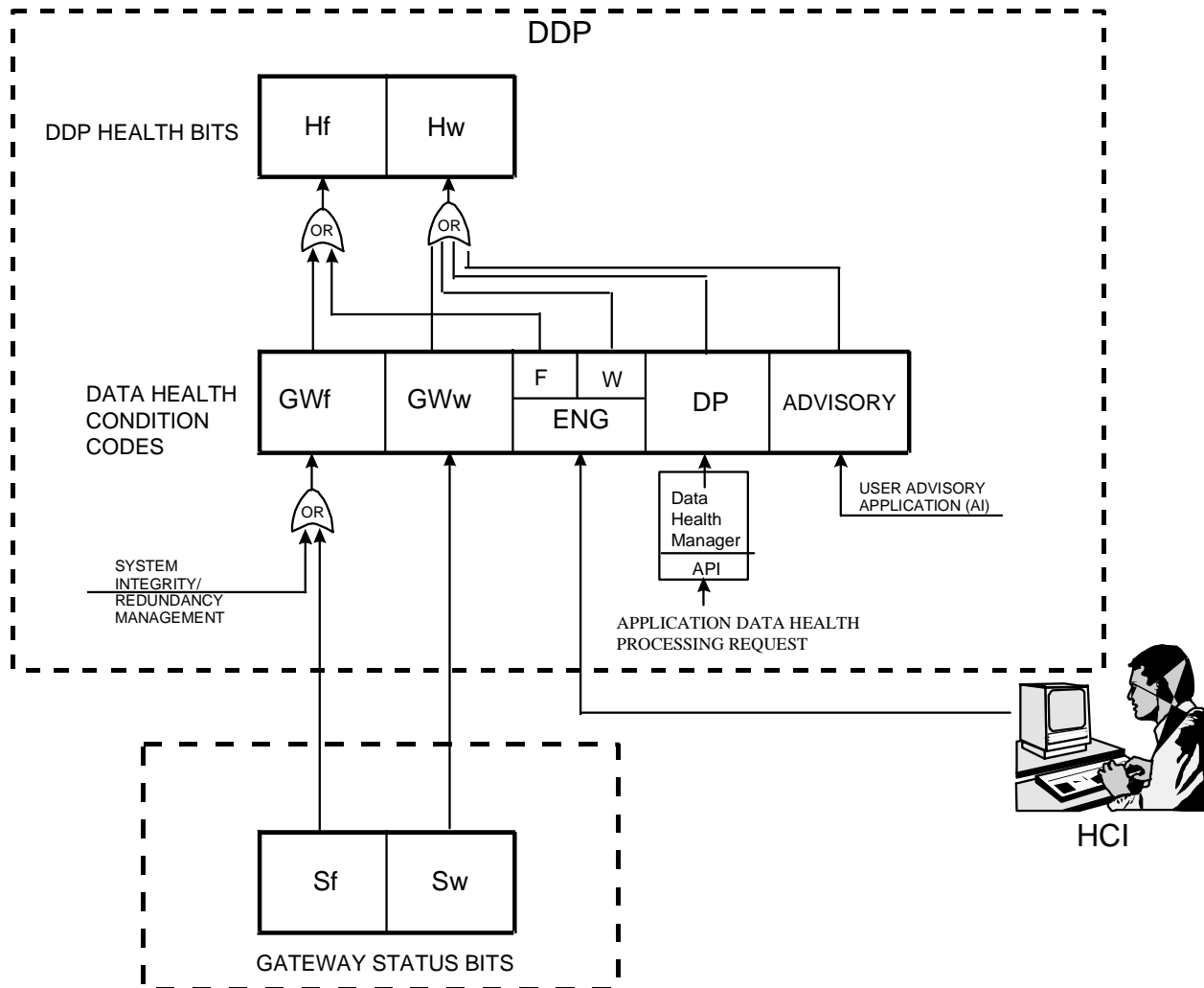
Data Health is the term applied to the integrity and validity of a Function Designator (FD) value which is being distributed from a CLCS subsystem. The health of an FD may represent the state of a hardware component, a communication path, resultant data from a Fusion operation, or application data. The boundaries of the data health measurement are the sending of the data by a component at one end of the data path and the receiving of data by a component at the other end. The status of the Data Health bit, as valid or invalid, determines the usability of the data.

The Data Health CSC is the software component that applies health status to each FD processed by the DDP, and provides the health status to applications at the HCI and CCP via Data Distribution.

Data Health CSC Overview is as follows:



Data Health Processing Concept



Each FD received from the gateway is accompanied with status described by two bits, a failure bit (Sf), and a warning bit (Sw). When the failure bit is set, it indicates that the FD value should be ignored. When the warning bit is set, it indicates that the validity of the FD is questionable, and that an operator or engineer intervention/decision may be needed.

The following examples of conditions that can set reason codes:

- Set to failure condition (ENGf) or warning condition (ENGw) by an engineer.
- Set to warning condition (DP) by the Data Health Manager based on application request.
- Set to warning condition (ADVISORY) by an Artificial Intelligence Application.

After any additional condition is applied to an FD, the Data Health Manager will compute the resultant health of the FD based on the gateway status and the current health conditions of the FD. Computation is done by OR'ing the Gateway failure bit with the failure conditions, and OR'ing the Gateway warning bit with the warning conditions. The resultant health, which will be represented by two Health bits (Hf and Hw), will be incorporated into Data Distribution, which will subsequently be distributed to all platforms, and made available for application access via FD Services.

1.1.2 Data Health CSC Operational Description

Data Health can be divided into two segments, a user development segment, and a run time segment.

The user development segment involves:

- User off-line definition of Data Health Algorithms via the use of a Data Path Editor. (To be reviewed for future delivery)
- Integrate Test build of the Data Health Algorithm Tables. (To be reviewed for future delivery)

The run time segment consists of :

- Loading the Data Health Algorithm Tables during DDP initialization. (To be reviewed for future delivery)
- Applying data health on FDs received at the DDP by the Data Health Manager. (Redstone)
- Storing data with health information in the CVT and making the data accessible by application via Data Distribution. (Redstone)

1.2 Data Health CSC Specifications

1.2.1 Data Health CSC Ground Rules

1. Data Health CSC will make use of Data Distribution API to obtain and store health information from/to the CVT.
2. Data health information will be made available for applications access at the DDP, HCI, and CCP via Data Distribution.
3. Applications, with exception of Data Health Manager and Data Fusion Manager, will obtain data health information via FD Services.
4. Goal is to use COTS tool selected for Fusion Editor to support Data Health requests.

1.2.2 Data Health CSC Functional Requirements

The Data Health Manager functional requirements are run time requirements to be implemented for the Redstone Delivery. For completeness, a list of functional requirements for the Data Path Editor, which will be used as specification for COTS evaluation, are also listed in Appendix D for reference.

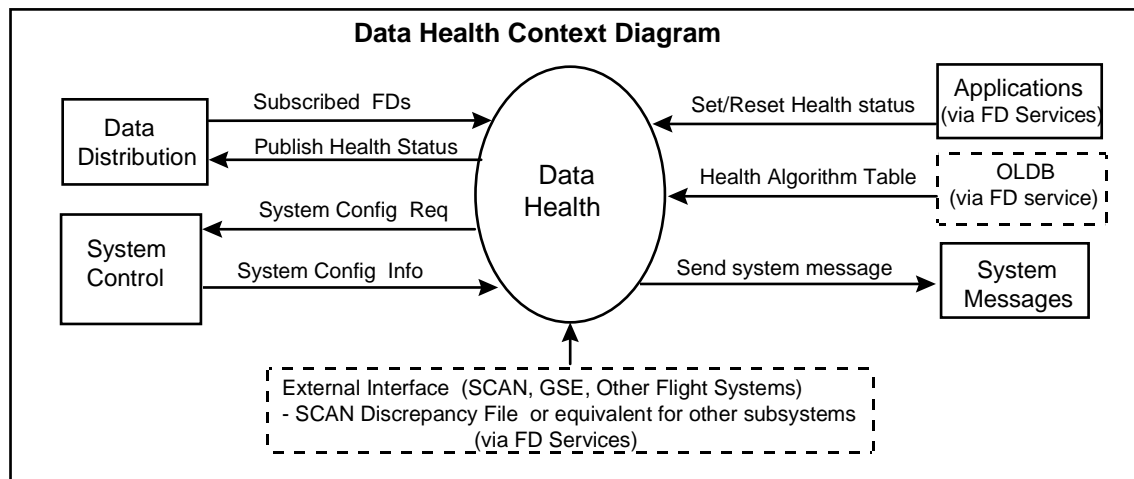
1.2.2.2 Data Health Manager (DHM) Functional Requirements

1. DHM will provide an application interface allowing applications to request Data Health processing by specifying the name of an FD and the Data Path health condition associated with the FD. DHM will provide an application interface allowing applications to request setting the Data Path condition code of an FD or a list of FDs to specific value(s), compute the resultant health, and incorporate the result into the Data Distribution CVT.
2. DHM will provide an application interface allowing applications to request Data Health processing by specifying the Data Path health condition, and a FD list ID, which represents a list of FDs defined in a FD algorithm file. DHM will provide an application interface allowing applications to request setting the Data Path condition code of a list of FDs to a specific value, via reference of an FD list ID, then compute the resultant health and incorporate the result into the Data Distribution CVT.
3. DHM will provide the capability to receive processing requests.
4. Upon receipt of a processing request, DHM will compute the resultant health based on status received from the gateway, the current Data Health conditions, and the input Data Path Health condition.
5. DHM will make the resultant health available for incorporation into Data Distribution.
6. DHM will provide an application interface for applications to issue a Data Health reason code request for an FD or a list of FDs.
7. DHM will provide the capability to receive Data Health reason code requests.
8. DHM will provide the capability to send the reason code(s) for an FD or a list of FDs to the requesting application.

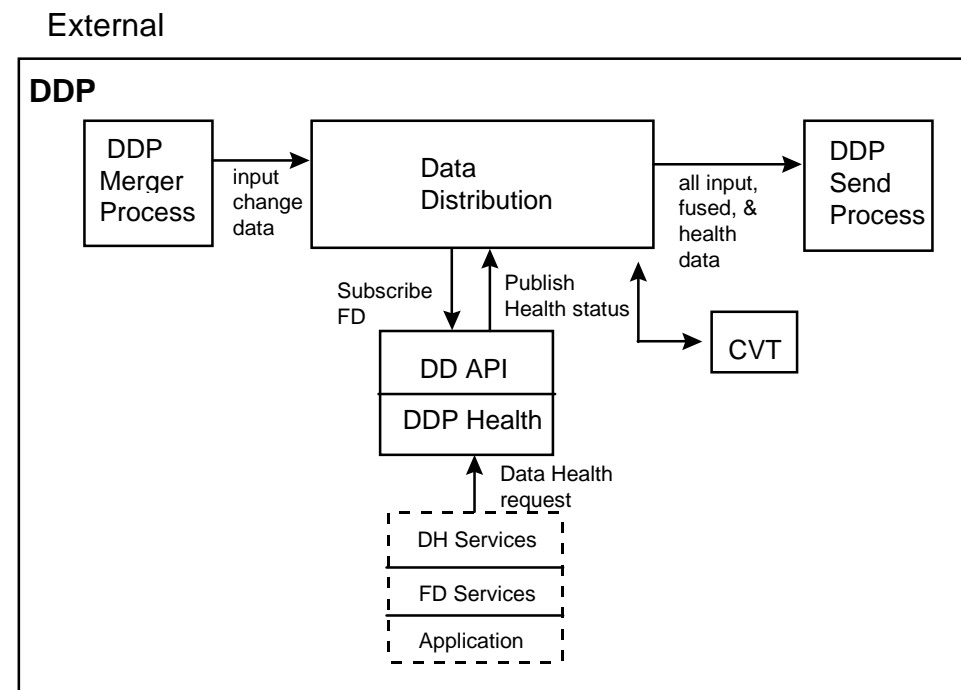
1.2.3 Data Health CSC Performance Requirements

The Data Health Function will process up to the “system maximum data bandwidth” of FDs.

1.2.4 Data Health CSC Interfaces



1.2.5 Data Health CSC Data Flow Diagram



1.3 Data Health CSC Specifications

The Data Health CSC applies health status to each FD processed by the DDP and provides the health status to applications at the HCI and CCP via Data Distribution.

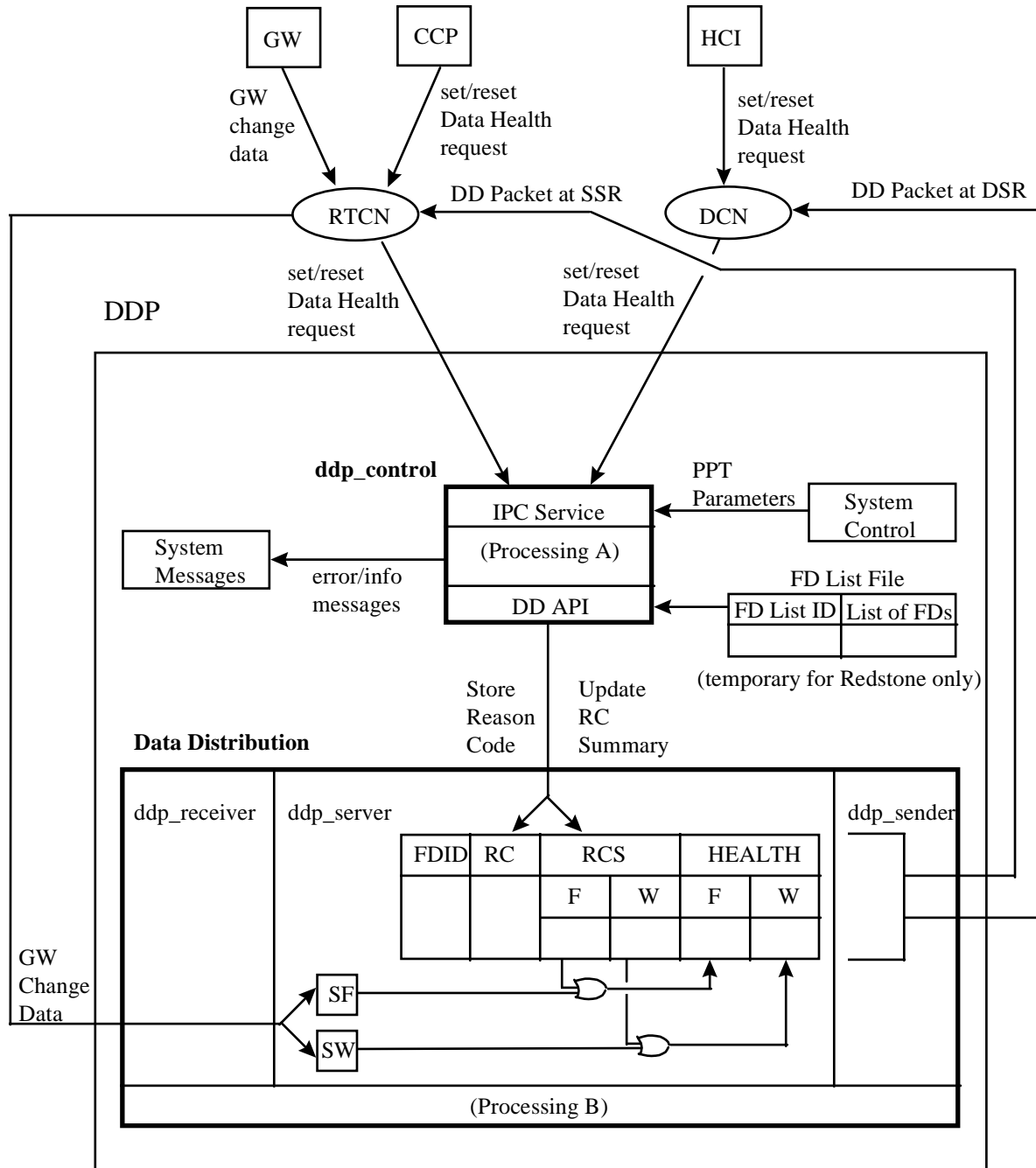
The Data Health CSC on the DDP will perform these functions:

1. Receive set/reset Data Health requests associated with an FD or a list of FDs from application residing on the DDP, CCP, and or HCI platforms.
2. Process the resultant health of FDs based on the gateway status and the current health conditions (same as reason codes) of the FD(s).
3. Incorporate the resultant health into the CVT via the use of Data Distribution API. The resultant health will then subsequently be distributed by Data Distribution to all platforms and made available for application access via FD Services.

The Data Health CSC on the DDP/CCP/HCI will provide application interfaces allowing applications to request set/reset Data Health and get Data Health reason codes.

1.3.1 Data Health Detailed Data Flow

1.3.1.1 DDP Detailed Data Flow



The Data Health CSC processing are performed partly in the ddp_control process and partly by Data Distribution.

Processing A

- The ddp_control process receives all C-to-C messages from all Data Distribution & Processing CSCs at the DDP. (For Redstone, there are only Data Health CSC C-to-C messages).

For Data health messages

- If request type is FD list ID, a list of FD names will be obtained from a FD list file
- Calls Data Distribution API to store input condition code to the CVT and compute reason code summary

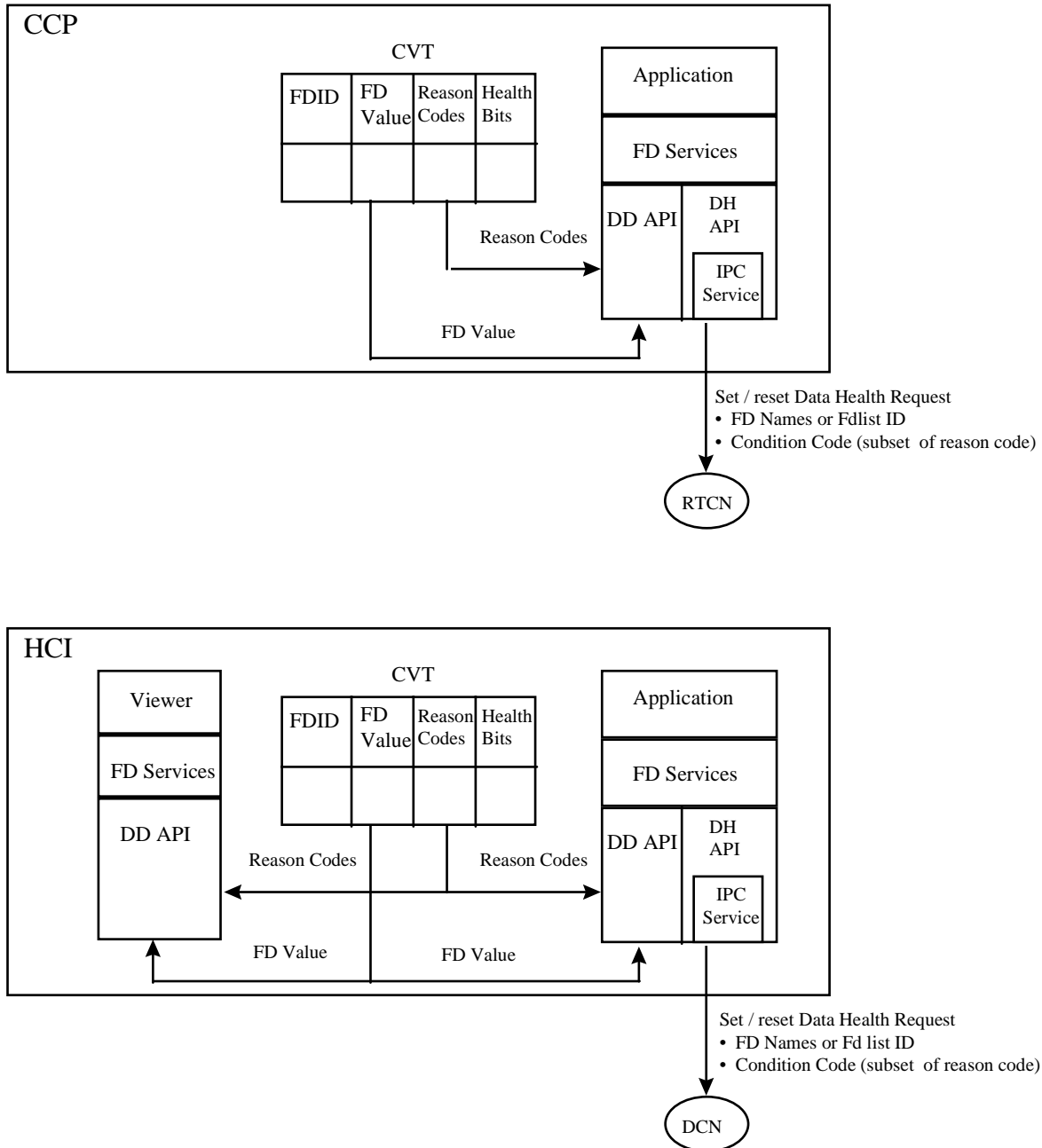
Processing B

- Data Health Processing performed by Data Distribution includes:
 - Receive gateway status bits as change data
 - Whenever there is a change in gateway status bit, or in reason code summary
 - ‘OR’ the gateway status bit with the corresponding reason code summary (failure to failure, warning to warning)
 - Store result as corresponding health bit in CVT
 - Output changed Health bits and reason code in the Data Distribution packet to RTCN (at SSR) and DCN (at DSR).

1.3.1.2 CCP/HCI Detailed Data Flow

On the CCP and HCI, applications use FD services, which in turn calls the Data Health API to request set/reset Data Health bits. The request is sent over the Network to the DDP via IPC Service.

On the HCI, the Viewers use FD Services, which in turn calls the Data Distribution API to get Data Health reason code from the local CVT.



1.3.2 Data Health External Interfaces

1.3.2.1 Data Health Message Formats

1.3.2.1.1 ddp_control messages

Message Number = 1

Message Group = DDP

Severity = Informational

DDP Control is initialized

Help Information Content:

The control process has initialized successfully

Detailed Information:

n/a

Message Number = 2

Message Group = DDP

Severity = Critical

DDP Control was unable to establish a connection with the server, errno = #ARGUMENT1#

ARGUMENT1 = unsigned integer representing UNIX error number value

Help Information Content:

During initialization of the DDP services, there was a problem creating a socket connection between the ddp_server and the ddp_control.

Detailed Information:

n/a

Message Number = 3

Message Group = DDP

Severity = Critical

DDP Control was unable to publish data to the server, errno = #ARGUMENT1#

ARGUMENT1 = unsigned integer representing UNIX error number value

Help Information Content:

During the processing of a user input request, DDP_control could not publish data to the ddp_server.

Detailed Information:

Make sure the ddp_server is still active.

Check if queue is full.

Message Number = 4

Message Group = DDP

Severity = Critical

DDP Control was unable to register with IPC Service, es_errno = #ARGUMENT1#, errno = #ARGUMENT2#

ARGUMENT1 = IPC services error number value

ARGUMENT2 = unsigned integer representing UNIX error number value

Help Information Content:

DDP control failed to register with IPC Service

Detailed Information:

See IPC Service error number for information

Message Number = 5

Message Group = DDP

Severity = Critical

DDP Control was unable to receive events from IPC Service, es_errno = #ARGUMENT1#, errno = #ARGUMENT2#

ARGUMENT1 = IPC Service error number value

ARGUMENT2 = unsigned integer representing UNIX error number value

Help Information Content:

DDP control failed to receive incoming events from IPC Service

Detailed Information:

See IPC Service error number for information

Message Number = 6

Message Group = DDP

Severity = Major

DDP Control was unable to deregister from IPC Service, es_errno = #ARGUMENT1#, errno = #ARGUMENT2#

ARGUMENT1 = IPC service error number value

ARGUMENT2 = unsigned integer representing UNIX error number value

Help Information Content:

DDP control failed to deregister from IPC Service

Detailed Information:

See IPC Services error number for information

Message Number = 7
Message Group = DDP
Severity = Informational

DDP Control is terminating

Help Information Content:
n/a

Detailed Information:
n/a

1.3.2.2 Data Health Display Formats

There are no display formats for the Data Health CSC.

1.3.2.3 Data Health Input Formats

There are no input formats for the Data Health CSC.

1.3.2.4 Data Health Recorded Data

There are no recorded data for the Data Health CSC.

1.3.2.5 Data Health Printer Formats

There are no printer formats for the Data Health CSC.

1.3.2.6 Data Health Inter-process Communication

1.3.2.6.1 IPC Services

Refer to IPC Services CSC for a description of the IPC header format. The Data Health request/response message body format to be transferred by IPC Services is depicted below:

Request Type 1 = Set health by FD Name(s) 2 = Set health by FD List ID
FD Count
FD Name / FD List ID
Condition Code
FD Name
Condition Code
FD Name
Condition Code
<ul style="list-style-type: none">•••

1.3.2.7 Data Health External Interface Calls

1.3.2.7.1 Data Health Interfaces

Allows the application to request Data Health processing of the specified FD or list of FDs, Data Path health condition associated with the FD(s). Also allows the calling application to reference a FD list ID, which represents a list of FDs defined in a FD list file.

- **ddh_set_health ()**

Inputs:

- Request type 1= request by FD name(s)
2 = request by FD list ID
- FD name(s) or FD list ID
- Condition code (s)

Output:

- Return code indicating success or failure. Error number if return code indicates failure.

- **ddh_get_reason ()**

Allows the application to retrieve the associated reason code(s) for the specified FD or list of FDs.

Inputs:

- FD name(s)

Output

- Reason code(s)

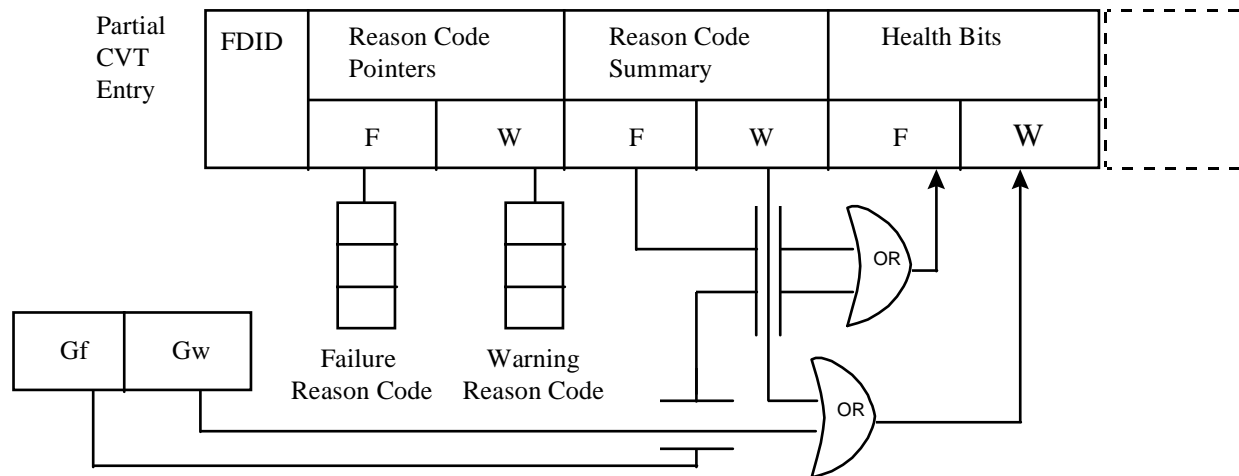
1.3.3 Data Health Internal Interfaces

1.3.3.1 Health Information in CVT

The health information for each FD is stored in the CVT entry. Information includes:

- Reason code (one for failure condition, one for warning)
- Reason code summary (summary = ON if one or more reason code exist, otherwise summary = OFF)
- Health Bits (one for failure condition, one for warning)

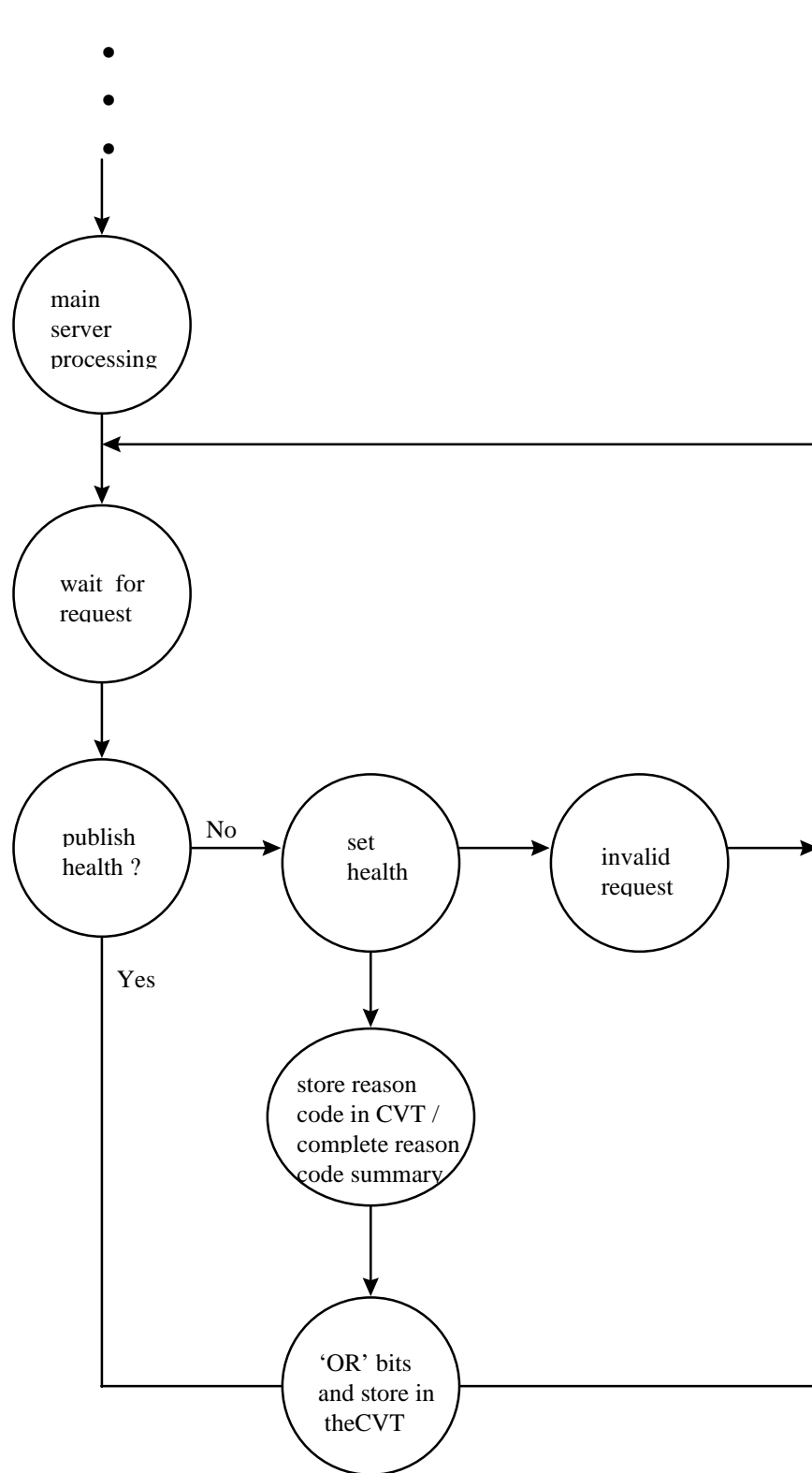
Whenever the Data Health Table is updated via the Data Health APIs or the DDP APIs, the gateway status bit for each entry is OR'ed with the corresponding reason code summary. The result of the operation is the Data Health of each FD, which will be stored into the CVT.



1.3.4 Data Health Structure Diagram

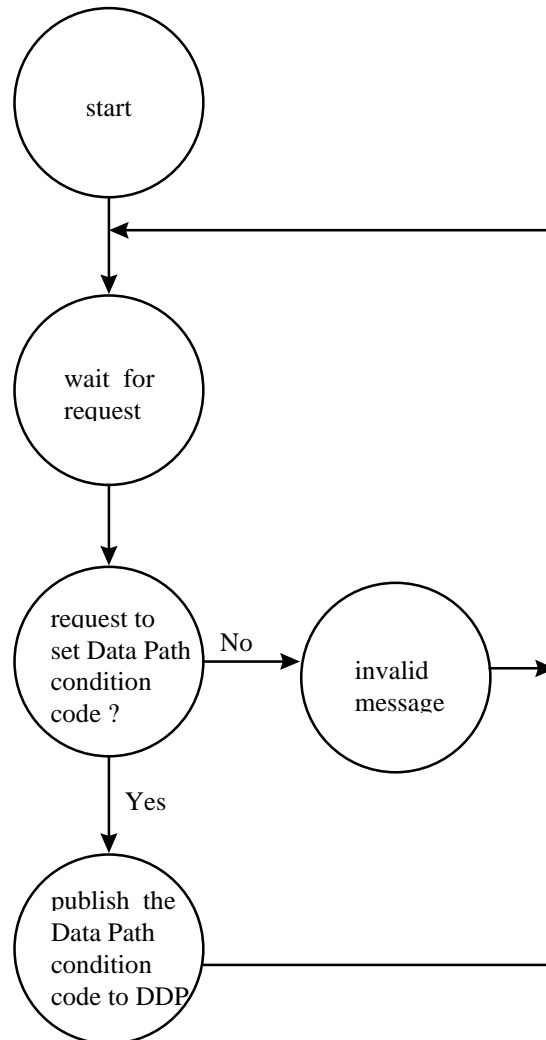
1.3.4.1 Server (DDP CSC)

The following diagram depicts the process performed by the Data Distribution ddp_server process in support of the Data Health processing request.



1.3.4.1.1 Data Health Control

1. The control process receives request via IPC Service from DDP/CCP/HCI.
2. The control process publishes the application specified Data Path condition code and/or reason code to the ddp_server.



1.3.5 Data Health Test Plan

1.3.5.1 Environment (IDE)

1.3.5.2 Test Tools

1. Data Generator - used to output up to 8 data streams to the DDP
2. PC Goal - used to compare data values with values on the HCI
3. SDC - used to validate data off-line
4. Data Distribution - used to maintain the CVT
5. Health Viewer - used to display health reason codes
6. IPC Service - used to send health requests from DDP/CCP/HCI to the DDP
7. Data Health Function - used to process up to the “system maximum data bandwidth” of FDs

1.3.5.3 Test Cases

1.3.5.3.1 Control

1. Verify the ddp_control can register with IPC Services.
2. Verify the ddp_control can establish a connection with ddp_server.
3. Verify the ddp_control can receive events from IPC Services.
4. Verify the ddp_control can publish to the ddp_server.

1.3.5.3.2 Health APIs

DD Health API testing will be accomplished by the development of simple code sequences that will exercise the ddh_set_health() and ddh_get_reason() calls.

Performance Testing

Performance testing for Data Health will be done as part of Data Distribution performance testing.

APPENDIX A

Statement of Work

- Develop the Concept of Operations of how Data Health will be utilized by the User.
 - Define the list of Data Health requirements for both system processes and user processes.
 - Define the relationships between Data Health and other system processes such as Data Fusion.
- Define and then provide the Initial Pre-build Data Path Health Editor. The viewer will, for all health bits, display any reason information including data path algorithms with input with multiple FD's.
- Define and then provide the Initial System Viewer capability for FD Health. The viewer will, for all health bits, display any reason information including data path algorithms with input and internal values. The viewer will allow updates at a TBD request rate.
- Define the Databank impacts for Data Health and provide the capability to add Data Health and Data Path Health information to the Databank.
- Confirm and/or modify system data flow for data health.
- Confirm and modify System Services for data health.
- Confirm and modify the Data Health Bits.
 - Provide for Gateway Provided status
 - Provide for DDP generation health bits.
 - Provide for advisory and engineering inputs from CCP and HCI
- Coordinate design with Data Distribution, and Data Fusion.
- Determine is a COTS tool can be utilized and implement the selected approach.
- Provide the capability to build Data Health Tables in the Test Build process.
- Incorporate Data Health information into Data Distribution.
- Provide Initial record and retrieval capability.
- Provide demonstration of at least one end to end GSE Analog health value.
- Provide performance data for system modeling.
- Provide the capability for the Data Health function to be utilized in both Operational and Application configuration.

APPENDIX B

Data Health Thread Deliverables

The target Data Health tasks for Redstone consist of the following:

- Provide assessment and recommendation on long term Data Health operations
- Find the best of class tool for Data Health run time processing and for Data Path Editor. Common tools that can support multiple functions will be highly desirable. For example, a tool that can be used both as the Data Fusion Editor and the Data Path Editor.
- Incorporate health bits into Data Distribution.

The following deliverable products will be provided for the Redstone Delivery:

- Source code and executable software that support:
 - Incorporation of gateway supplied health bits into Data Distribution.
 - API to support application requests to set/reset specific health bit(s).
 - Interface between DDP and Data Health Manager.
- Data Path Editor specifications.
- Data Health run time specifications.
- COTS evaluation report to provide recommendation based on weighed criteria.

If an acceptable tool is found, the following will be delivered into the Application Test Bed environment:

- The COTS tool.
- Documentation provided by the vendors.
- Provide the capability to monitor the state of a predefined list of FDs against their predefined required states. When one or more fail to match, set the Data Path bit.
 - The list of FDs and the target state, will all be pre-defined in a manually generated file for Redstone. This file will be generated by the Data Path Editor for later deliveries.

The following will be implemented for post Redstone deliveries:

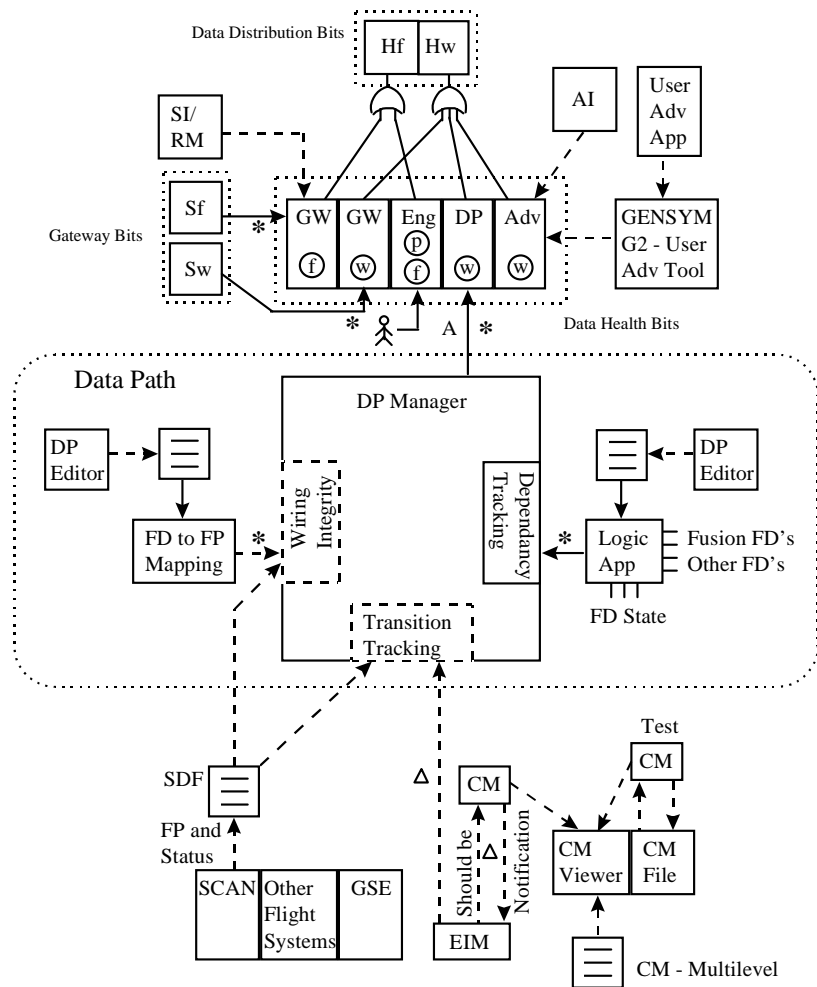
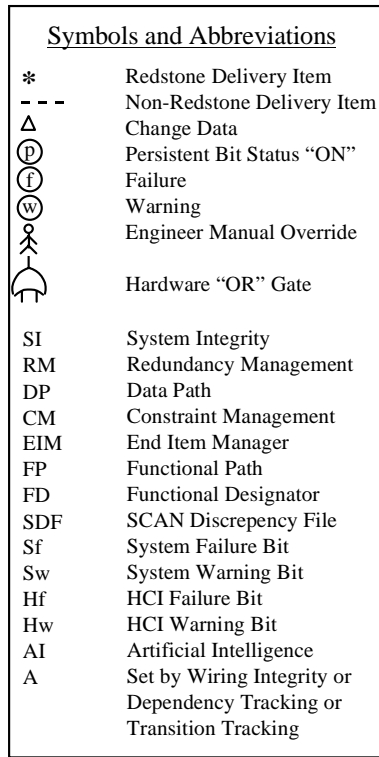
- Parse SCAN Discrepancy File and set the Functional Path associated FDs accordingly.
 - SCAN file will be parsed once per shift.
 - SCAN file will be manually loaded once per shift
 - The Functional Path/FD Mapping File will be generated by the Data Path Editor.
- Provide support to parse or analyze data generated by GSE and other Flight Systems, then set the Functional Path associated FDs accordingly
- Provide APIs to support the following interface:
 - Transition Tracking
- The list of FDs and the target state for Dependency tracking will be generated by the Data Path Editor.

The future requirements for Data Health, which include but are not limited to the following support, are not covered in this document:

- Parse SCAN Discrepancy File and set the Functional Path associated FDs.
- Parse and analyze data generated by GSE and other Flight Systems, then set the Functional Path associated FDs.
- Realtime monitoring of the state of a pre-defined list of FDs against their predefined required states. When one or more fail to match, set the Data Path bit.

APPENDIX C

Data Path Health Concept



APPENDIX D - Data Path Editor Specifications

1. DPE will provide an Graphical User Interface (GUI) allowing user to perform these functions:
 - Select input FDs using a pre-defined list of valid FDs
 - Select support options
 - Wiring Health
 - Transition Tracking Health
 - Dependency Tracking Health
 - Define Functional Path and its associated FD (for Wiring Health option)
 - FD name
 - Functional Path Name
 - Define a FD or a list of FDs, whose Data Path bit will be set/reset, as requested by an EIM, when a specific constraint violation has occurred. (for Transition Tracking option)
 - FD name list
 - User specified FD list ID
 - Define a list of FDs and an associated FD, whose health status will be affected by the state of one or more of the FDs on the list (Dependency Tracking)
 - FD name
 - FD name list
 - Required states/limits
 - Units
 - Enter Comments
 - Modify or delete the algorithm defined for an existing FD
 - Save, save as, exit without saving the contents of the definition in a file
 - Print screen
2. DPE will provide On-Line Help
3. DPE will generate the following files upon user selection:
 - Functional Path (FP)/FD association file
 - Transition Tracking FD file
 - FD State file

ok

▲

▼

Data Path FD Name

Refresh

Execute

Print

Cancel

ok

▲

▼

Functional Path

State

ok

▲

▼

FD Name

FD Name List	Required States/Limits	Units

ok

▲

▼

FD Name

FD Name List	Value	Units

Data Path Editor - Text Description

The Data Path Editor (DPE) provides a graphical user interface allowing users to update the data path of a FD. Adding allows for the creation of a new data path for a FD. Changing allows for the modification of an existing data path for a FD. The deleting and querying (viewing without changing) functions will not be supported in this initial data path editor. Printing allows for the producing of a printed copy of a new or existing data path for a FD.

The DPE screen is divided into four areas: FD Selection, Wiring Health, Dependency Health, and Transition Health. The FD Selection area contains a Data Path FD Name Box and four command buttons. The name of the data path FD can be manually entered or selected from a scrolling list in the Data Path FD Name Box. The list of names is scrolled by clicking on the up and down arrows. (The scrolling feature for the data path FD names will need to be improved because of the large number of FDs, but for the Redstone delivery only a small set of FDs will be used.) After the correct name is displayed in the data path FD box, the OK box is clicked to accept the name. The Refresh button will redisplay the screen, disregarding any updating of the data path for the FD. The Execute button will replace the previous data path for the FD with the updated settings. The settings include the data path FD name, wiring health, dependency health, and transition health. The Print button will produce a printed copy of the data path for the FD settings. The Cancel button will leave the DPE screen.

The Wiring Health area contains a Functional Path Box and a State Box. The name of the functional path can be manually entered or selected from a scrolling list in the Functional Path Box. The list of names is scrolled by clicking on the up and down arrows. (The scrolling feature for the functional path names will need to be improved because of the large number of paths, but for the Redstone delivery only a small set of paths will be used.) After the correct name is displayed in the data path FD box, the OK box is clicked to accept the name. The state of the functional path (Valid or Invalid) is displayed in the State Box.

The Dependency Health area contains a FD Name Box, a FD Name List Box, a Required States/Limits Box, and a Units Box. The name of the FD can be manually entered or selected from a scrolling list in the FD Name Box. The list of names is scrolled by clicking on the up and down arrows. After the correct name is displayed in the FD box, the OK box is clicked to accept the name. (The scrolling feature for the dependency names will need to be improved because of the large number of names, but for the Redstone delivery only a small set of names will be used.) The associated FD Name List, Required States/Limits, and Units can be updated and scrolled as a group and individually. The required states for discrete data items will be in the formats of Valid/Invalid, On/Off, Open/Closed, etc. The limits for analog data items will be in the format of Low : number value, number value : number value, number value : High, etc. The Units Box will show the measuring units used such as PSI, meters, degrees, etc.

The Transition Health area contains a FD Name Box, a FD Name List Box, a Value Box, and a Units Box. The name of the FD can be manually entered or selected from a scrolling list in the FD Name Box. The list of names is scrolled by clicking on the up and down arrows. After the correct name is displayed in the FD box, the OK box is clicked to accept the name. (The scrolling feature for the transition names will need to be improved because of the large number of names, but for the Redstone delivery only a small set of names will be used.) The associated FD Name List, Value, and Units can be updated and scrolled as a group. The values for discrete data items will be in the formats of Valid, Invalid, On, Off, Open, Closed, etc. The values for analog data items will be in the format of 100, - 365, 0.1234, etc. The Units Box will show the measuring units used such as PSI, meters, degrees, etc.